Using a High-Level I/O library for Improved Performance: ADIOS

Jay Lofstead

Scott Klasky, Norbert Podhorszki, Qing ‘Gary’ Liu

December 7, 2009

Overview

- Why ADIOS
- How to use it
- Compatibility

Motivation

- Multiple HPC architectures
  - Cray, IB-based clusters, BlueGene

- Many different APIs
  - MPI-IO, POSIX, HDF5, netCDF
  - GTC (fusion) has changed IO routines 8 times so far based on moving platforms

- Different IO patterns
  - Restarts, analysis, diagnostics
  - Different combinations provide different levels of I/O performance

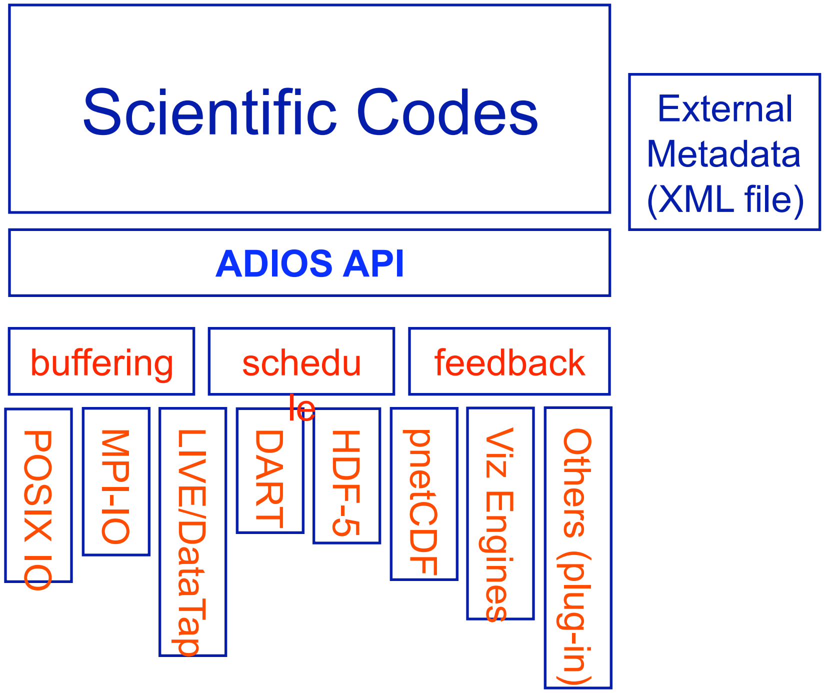
Design Goals

- ADIOS Fortran and C based API almost as simple as standard POSIX IO
- External configuration to describe metadata and control IO settings
- Take advantage of existing IO techniques (no new native IO methods)

Fast, simple-to-write, efficient IO for multiple platforms without changing the source code

Architecture

- Thin API
- XML file
  - **data groupings** with annotation
  - **IO method selection**
  - buffer sizes
- Common tools
  - Buffering
  - Scheduling
- Pluggable IO routines



Scientific Codes

External Metadata (XML file)

ADIOS API

buffering

schedule

feedback

POSIX IO

MPI-IO

LIVE/DataTap

DART

HDF-5

pnetCDF

Viz Engines

Others (plug-in)

Supported Features

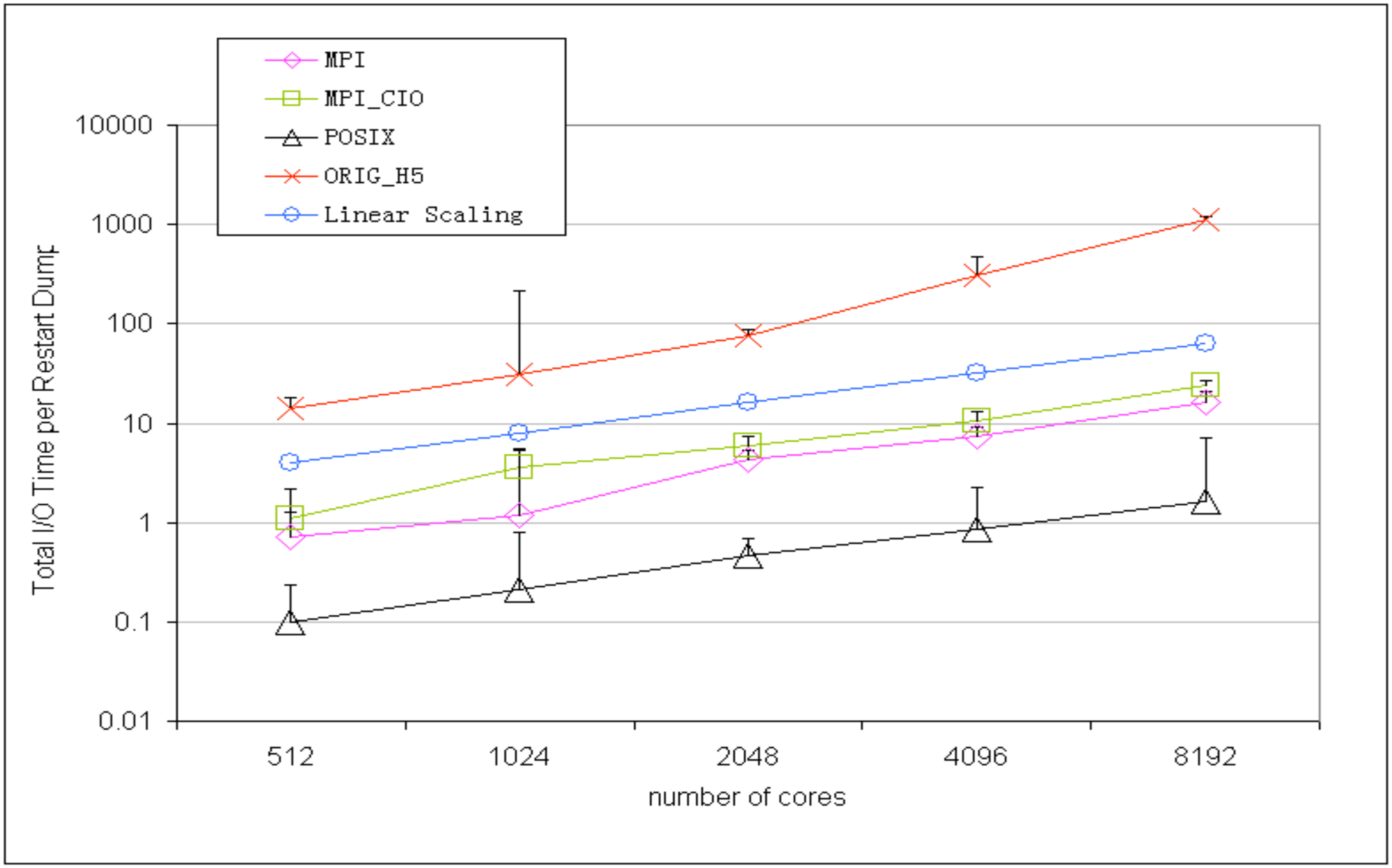
- Platforms tested
  - Cray CNL (Jaguar, JaguarPF)
  - Cray Catamount (old-Jaguar and SNL Redstorm)
  - Linux Infiniband (Ewok)
  - BlueGene/P (Eugene)
  - MacOS (limited support)
- IO Methods
  - MPI-IO (general and Lustre optimized), HDF5, POSIX, NULL
  - Ga Tech DataTap asynchronous, Rutgers DART asynchronous

Performance!

- Chimera Supernova
  - Restarts: 1 MB/proc, weak scaling
- GTC Fusion
  - Particles only: 11.5 MB/proc, weak scaling
  - Restarts: 116.5 MB/proc

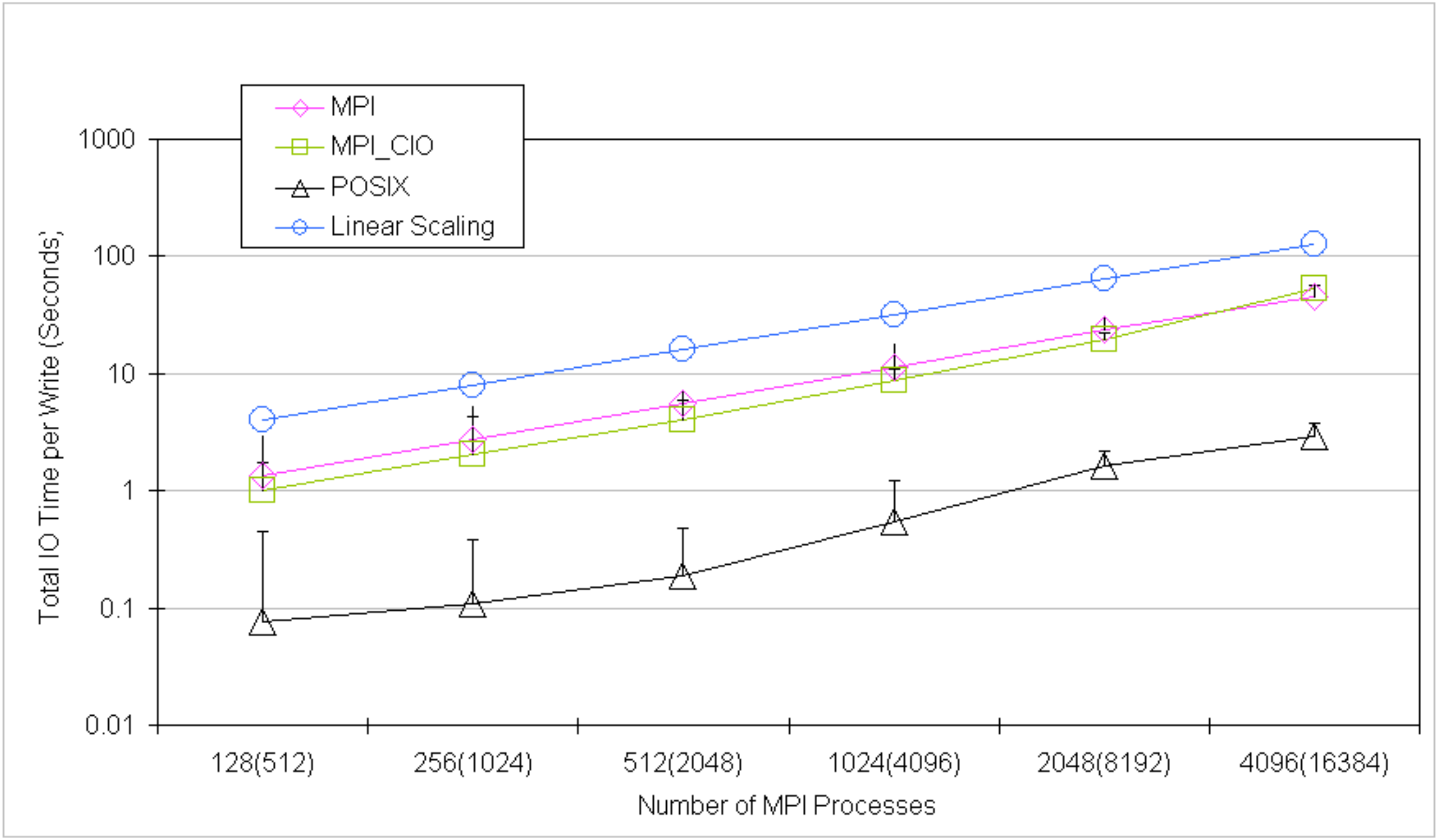
Chimera on Jaguar

- 1 MB/proc



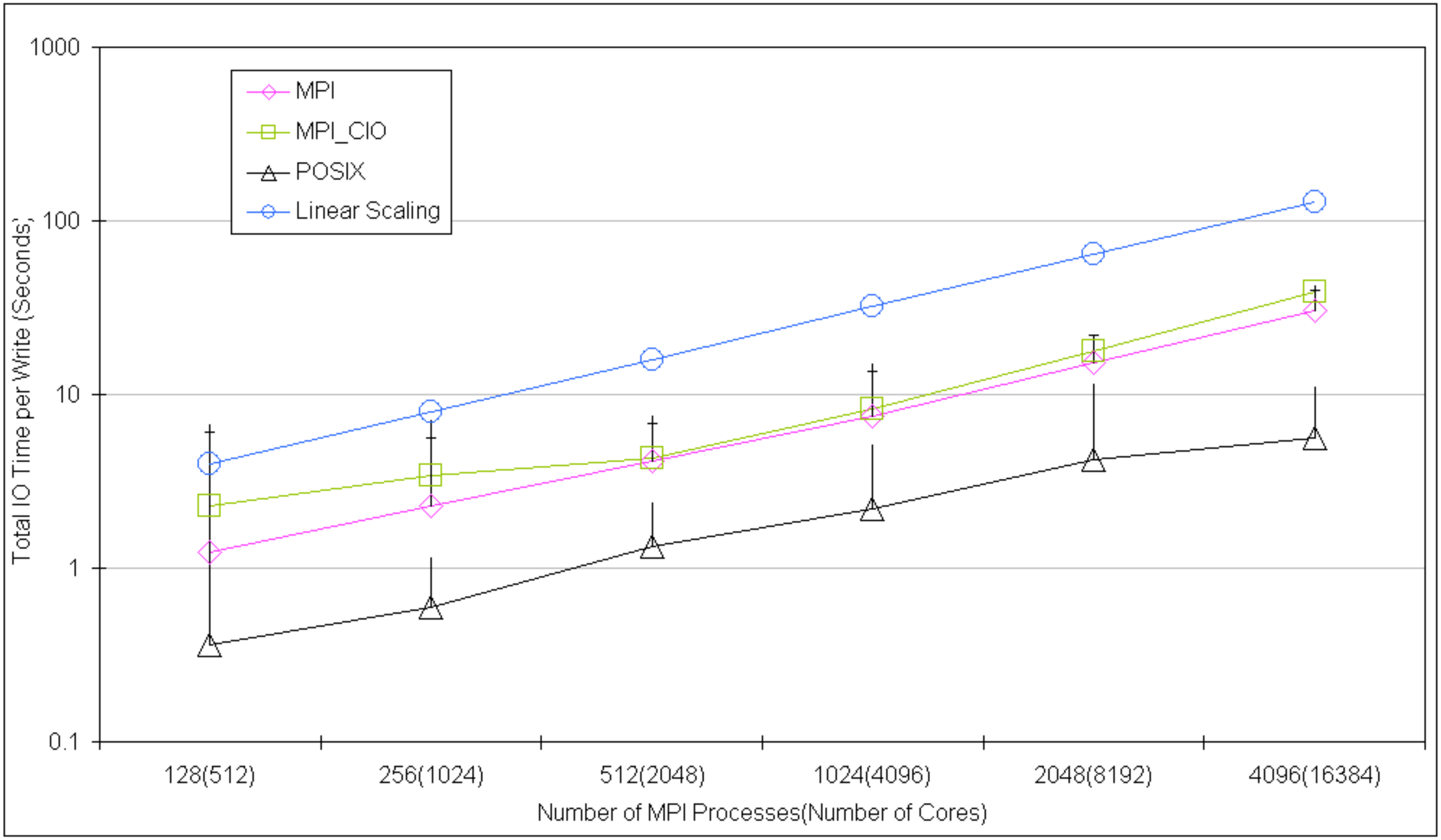
GTC Particles Weak Scaling

- 11.5 MB/MPI process

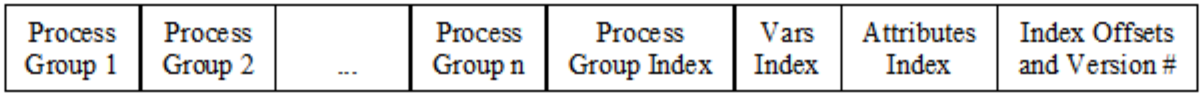


GTC Restarts Weak Scaling

- 116.5 MB/MPI process



BP File Format



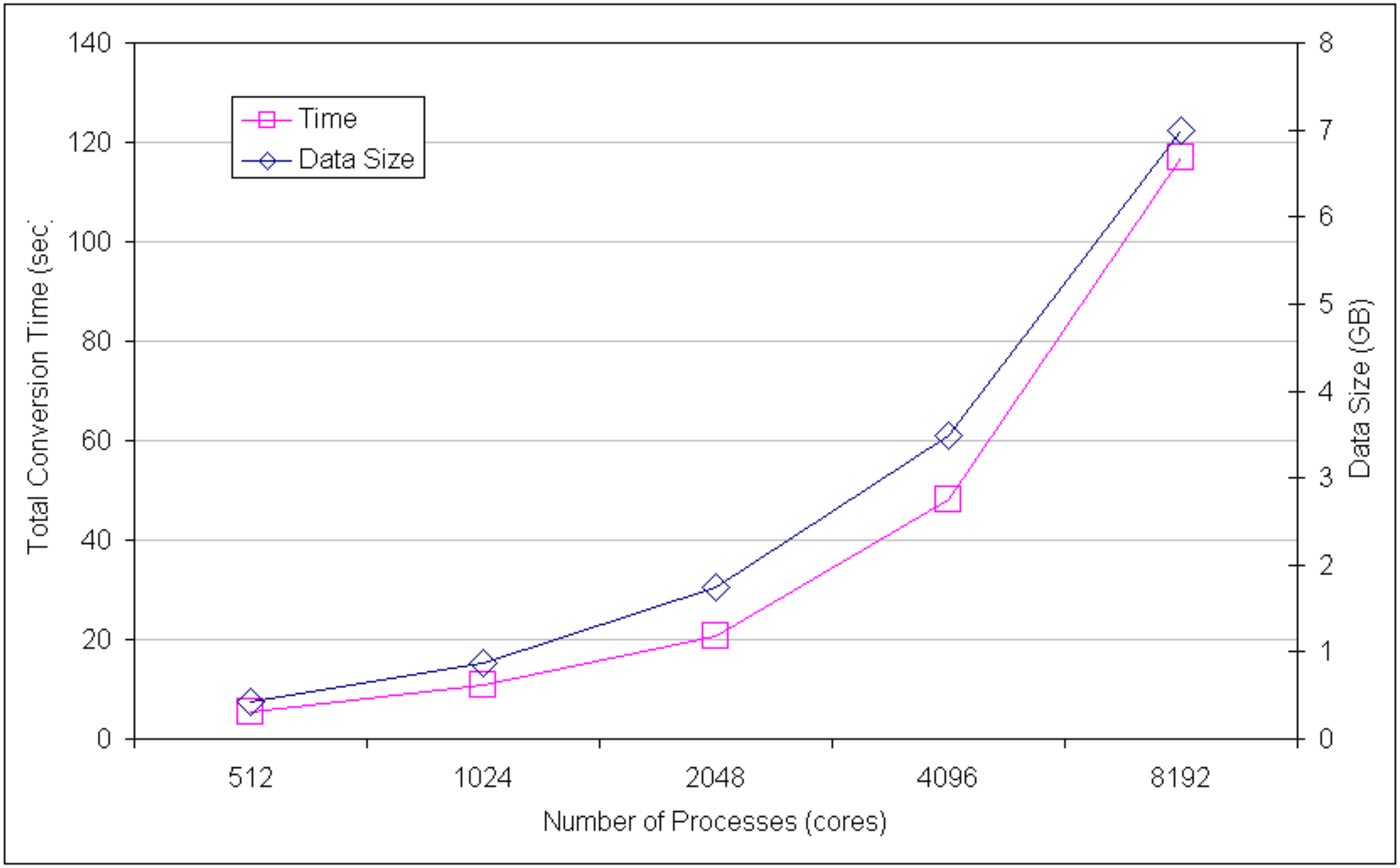
Process Group 1	Process Group 2	...	Process Group n	Process Group Index	Vars Index	Attributes Index	Index Offsets and Version #

- Failure of single writer (even root) not fatal
- Each process has separate area to write
- Essentially a superset of NetCDF and HDF-5 for each process group with an overall index
- All data characterized
- All data and output indexed automatically

- Primarily an intermediate format
- Fully 64-bit enabled

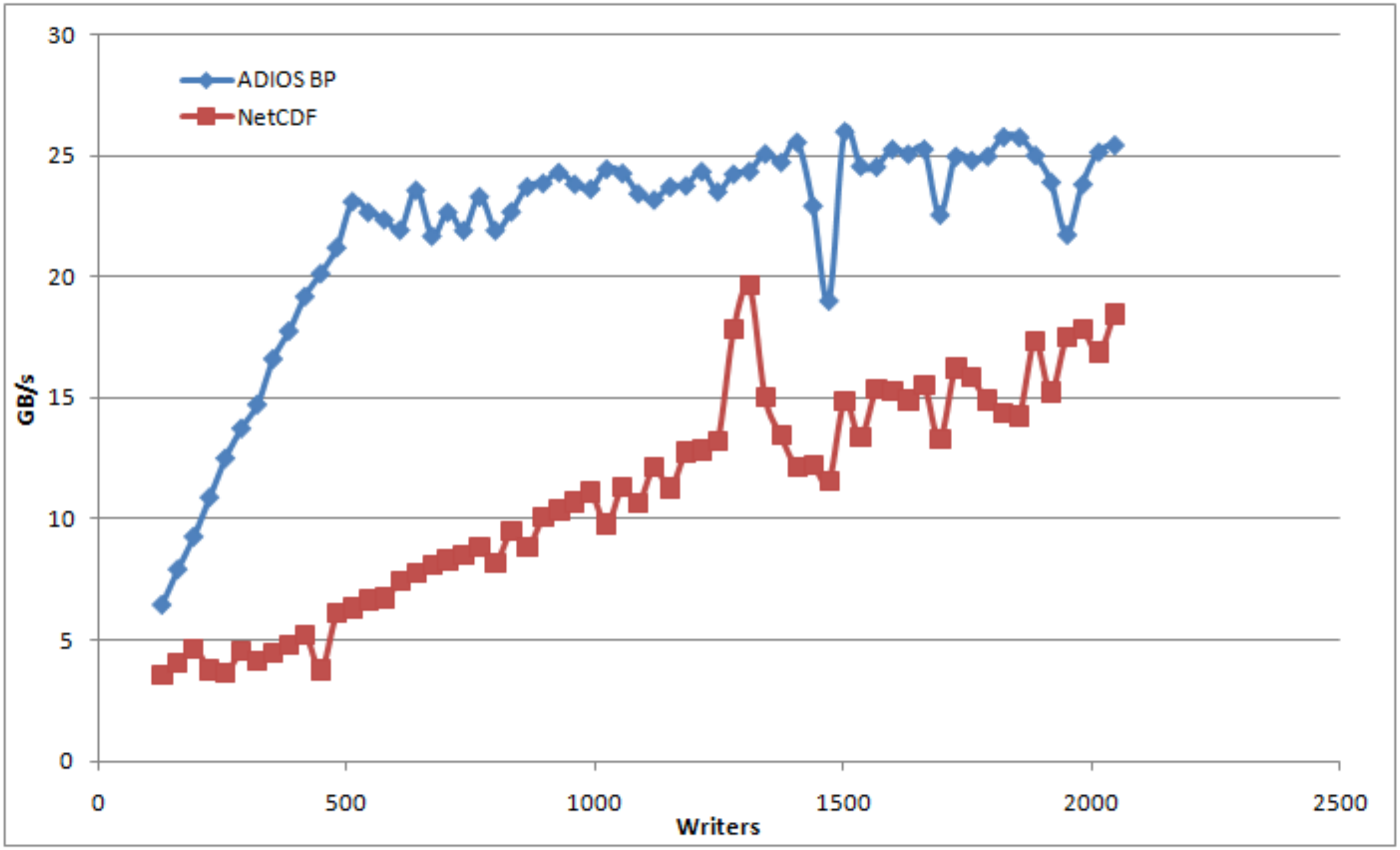
What About File Conversion?

- Single process used for conversion

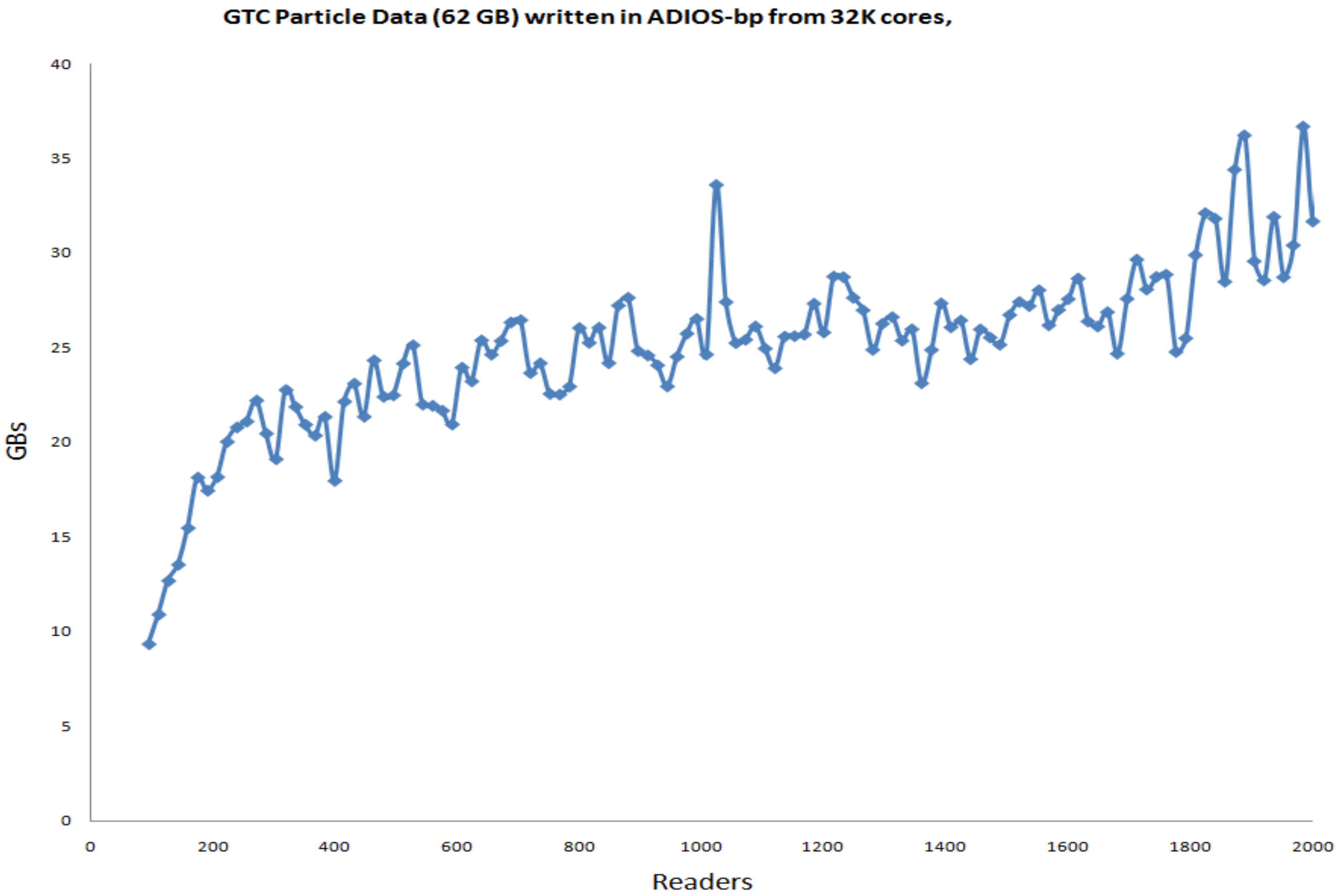


Read Performance?

- *Pixie3D Large Data Read on Half Procs*



Arbitrary Reads



How does the code look?

- Setup/Cleanup code

```
call adios_init (‘config.xml’)
...
call adios_finalize (my_rank)
```

- adios_init – reads the XML file (once from proc 0 and broadcast)
- adios_finalize – provide opportunity for cleanup

The General IO Routines

call adios_open (handle, ‘groupname’, ‘filename’, mode, communicator)

#include “groupname_write.fh”

call adios_close (handle)

Read version available as well!

(Python script ‘compiles’ XML file, with special markup, into includes for Fortran and C)

The Detailed IO Routines

```
call adios_open (handle, ‘groupname’,
   ‘filename’, mode, communicator)


call adios_group_size (handle, data, total)


call adios_write (handle, ‘varname’, var)
ADIOS_WRITE(handle,var_name)
…
call adios_close (handle)
```

IO Details

- Writing AND reading are buffered
- Coordination points are limited for greater independence
- Data characteristics and scalars can be read directly from index in constant time
  - includes mix/max for ALL vars no matter the file or data size!

What about that pesky XML?

- Describe each IO grouping
- Map an IO grouping to transport method(s)
- Define buffering allowance

‘XML-free’ API completed and in final testing

XML Overview

- XML file contents (data elements)

```
<adios-config host-language=“Fortran”>
<adios-group name=“restart”>
<var name=“elements” type=“integer”/>
<var name=“data” type=“double” path=“/”
   dimensions=“elements”/>
</adios-group>
</adios-config>
```

XML Overview

- XML file contents (other)

```
<attribute name=“description” path=“/data”
   value=“simulation particle data”/>

<global-bounds dimensions=“..” offsets=“..”>
<var .../>
</global-bounds>

<transport method=“MPI” group=“restart”/>

<buffer size-MB=“100” allocate-time=“now”/>
```

General Read Routines

- Open file
- Inquire file contents
- Open group
- Inquire group contents
- Inquire var info
- get var data
- close group
- close file

General Read Routines

adios_fopen (handle, ‘filename’, communicator)

adios_inq_file (handle, group_count, var_count, attr_count, time_start, time_stop, groupname_list)

adios_gopen (handle, ghandle, ‘name’)

adios_inq_group (ghandle, var_count, varname_list)

adios_inq_var (ghandle, ‘name’, var_type, var_rank, vartime_dim, dims)

adios_get_var (ghandle, ‘name’, buffer, start, readsize, time_start)

adios_gclose (ghandle)

adios_fclose (handle)

ADIOS Tools

- bpls
    - Similar to h5dump/ncdump
    - Also shows array min/max values
    - Performance independent of data size
- adios_lint
    - Validate the XML file
- bp2h5, bp2ncd
    - Convert BP format into HDF5 or NetCDF

Asynchronous IO Hints

call adios_end_iteration ()

- pacing hints
- use in conjunction with ‘iterations’ attribute of method element in XML

call adios_begin_calculation ()

- a low-IO phase is starting

call adios_end_calculation ()

- a low-IO phase is ending

Integrated Science Codes

- Fusion
  - GTC, GTS, XGC-1, XGC-0, M3D, M3D-K, Pixie3D
- Astrophysics
  - Chimera
- Combustion
  - S3D
- AMR Frameworks
  - Chombo
- Others
  - GEM, GTK

Platforms Supported

- Full functionality on Linux & BG/P
  - Includes full API and Matlab & VisIt integration
- Limited functionality on MacOS
  - Limited to general read API only
  - Matlab and VisIt read only
  - bpls and FIESTA plotter work

More Information

NCCS ADIOS webpage:

http://www.nccs.gov/user-support/center-projects/adios/

ADIOS Wiki (overview docs)

http://adiosapi.org/

ADIOS full documentation

Part of the download from NCCS

Acknowledgements

This work was funded by

- National Center for Computational Science, Oak Ridge National Labortatory
- Sandia National Laboratories under contract DE-AC04-94AL85000
- a grant from NSF as part of the HECURA program
- a grant from the Department of Defense
- a grant from the Office of Science through the SciDAC program
- the SDM center in the OSCR office